

Serial No. 10/021,080
Art Unit 2193

DEC 22 2006

REMARKS

As a preliminary matter, the Applicant takes this opportunity to point out that the Office Action Summary refers to claim 21 rather than claim 20. Claim 21 has been previously canceled, and claim 20 is still before the Examiner.

The Examiner has rejected claims 1-9, 12-20, and 22-24 as not complying with 35 U.S.C. 103(a) as being obvious in view of "The Benefits of CORBA-Based Network Management" by Haggerty and Seetharamann and in view of U.S. Patent 6,898,792 issued to Burke.

Haggerty teaches the use of CORBA in network management. The object hierarchy that is taught is the usual object hierarchy of object-oriented programming, as illustrated in Figure 4 and in the corresponding description in the paragraph bridging pages 76 and 77. All objects, there being more than one, are derived from one base object, in that all objects inherit some properties and functions of the base object such as a name. In the implementation of Haggerty, derived objects are defined for network equipment. Haggerty is talking about object types, as indicated by discussion of inheritance of properties and functions.

Actual instantiation of the objects of the types defined according to the generic network model occurs by explicit addition to the Managed Object Manager through the GUI, or by auto discovery. This is explained in the first full paragraph of page 76 of Haggerty, with reference to Figure 3. As is well known to those skilled in the art, auto discovery in a network management system context refers to automated polling of actual network elements in order to discover the existence or properties of the network elements.

In order to provide NMS services in a typical network, the method taught by Haggerty would require the definition of many types of classes, which would require providing an interface to clients that consist of hundreds of classes. As new types of equipment are added, this interface will grow. Changes to methods and attributes of these classes would usually require modification to the interface. Avoiding this large and costly changing of the interface is one object of the present claimed invention. A single class with

Serial No. 10/021,080
Art Unit 2193

a small, well defined set of methods is used. The methods and attributes of all the derived classes in the class hierarchy can be accessed through this small, immutable interface.

These differences will be demonstrated by considering the elements of the present claims.

Claim 1 is directed to a framework which includes a directive parser configured to process, at run-time, at least one self-contained managed data network entity specification file including directives. The Examiner has cited page 76 paragraph 1 of Haggerty as teaching this element, in particular the sentence "The topology objects are created through ... auto discovery". The Applicant respectfully submits that this passage of Haggerty does not teach a directive parser for processing, at run-time, at least one self-contained managed data network entity specification file including directives. As explained above, auto discovery is the process of automatically discovering the existence and properties of elements within a network. Haggerty is teaching the instantiation of objects in response to information learned through auto discovery of a network. This is clearly not parsing of network entity specification files. The Applicant also notes that the Examiner seems to refer to the HP OpenView Reference. This reference has not been cited against the present application, nor has the Examiner provided a copy of this reference to the Applicant.

Claim 1 also includes an executable code implementation of a single managed entity object class, the single managed entity object class being run-time derivable via type derivation into a derivation hierarchy of managed network object types based on run-time parsed entity derivation directives. It is important to note that this executable code implementation is not simply instantiating a number of objects within an existing class hierarchy, but rather is creating a hierarchy of object types at run-time. The Examiner has cited the paragraph bridging page 76 and 77 of Haggerty and Figure 4 of Haggerty as teaching this element. However, Haggerty is teaching a base class from which other classes are derived. This is a standard object-oriented class hierarchy, and Haggerty gives no indication that the class hierarchy is created at run-time. What happens at run-time, as discussed by Haggerty and by the Examiner, is instantiation of objects, not creation of object types. This difference means that interfaces to a large number of object types need not be provided to the client.

Serial No. 10/021,080
Art Unit 2193

Claim 12 includes similar limitations of method steps of parsing directives from at least one managed data network entity specification file, and deriving at run-time a single managed entity object class into a managed entity object type derivation hierarchy.

Claims 2-9 are dependent on claim 1 and include the same limitations discussed above. Claims 13-20 and 22-24 are dependent on claim 12 and include the same limitations discussed above. Because the Examiner has not shown where each and every element of claims 1-9, 12-20, and 22-24 are taught or suggested by Haggerty and Burke, either alone or in combination, the Applicant respectfully submits that a *prima facie* case of obviousness has not been established against the claims.

In view of the foregoing, it is believed that the claims at present on file are in condition for allowance. Reconsideration and action to this end is respectfully requested.

Respectfully submitted,



S. Mark Budd
Registration No. 53,880
Agent of Record

MARKS & CLERK
P.O. Box 957, Station B
Ottawa, ON K1P 5S7 (613)236-9561